



(19)

(11) Publication number: **03269'**

Generated Document.

PATENT ABSTRACTS OF JAPAN(21) Application number: **02070678**(51) Intl. Cl.: **G06F 7/52**(22) Application date: **20.03.90**

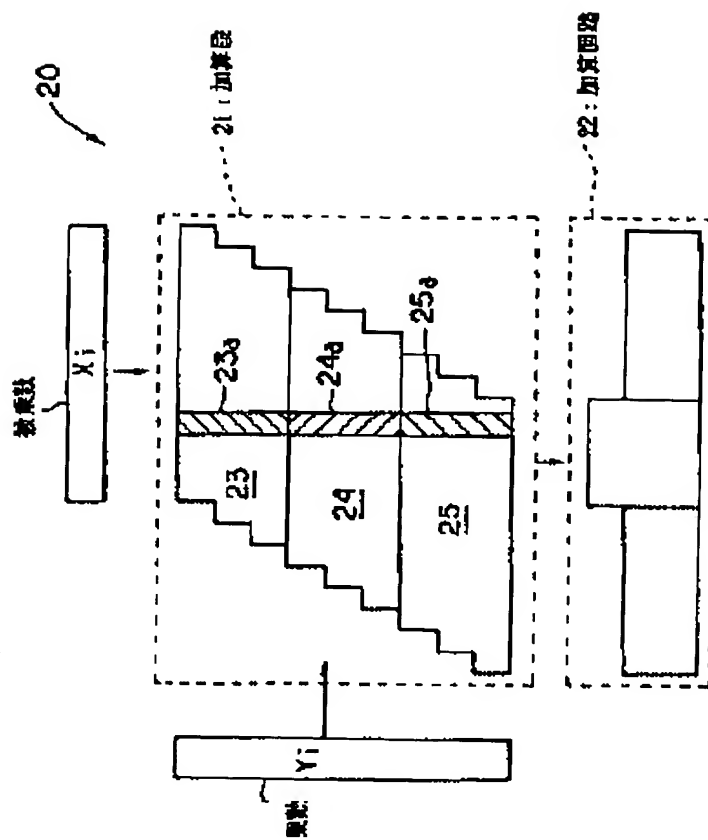
(30) Priority: (43) Date of application publication: 02.12.91 (84) Designated contracting states:	(71) Applicant: FUJITSU LTD FUJITSU MICROCOMPUTER LTD (72) Inventor: IINO HIDEYUKI HIDA HIDENORI (74) Representative:
--	---

(54) MULTIPLIER**(57) Abstract:**

PURPOSE: To increase the operation speed and to improve the easiness of layout by dividing an adding stage to plural stages and constituting each divided adding stage with an array type multiplier and providing an adding circuit having the Wallace tree constitution which adds the outputs of respective divided adding stages.

CONSTITUTION: An adding stage 21 whose number of operation stages corresponds to the number of the bits of a multiplier yi is divided to plural stages, and each of divided adding stages 23 to 25 consists of an array type multiplier, and an adding circuit 22 having the Wallace tree constitution is provided which adds the outputs of divided adding stages 23 to 25. When the division number of the adding stage 21 is properly set, the advantages of both the array type and the Wallace type are effectively used. Thus, the easiness of layout is improved and the operation speed is increased.

COPYRIGHT: (C)1991,JPO&Japio



⑩ 日本国特許庁(J P)

⑪ 特許出願公開

⑫ 公開特許公報(A) 平3-269724

⑬ Int. Cl.⁵
G 06 F 7/52

識別記号 庁内整理番号
3 1 0 M 2116-5B

⑭ 公開 平成3年(1991)12月2日

審査請求 未請求 請求項の数 1 (全7頁)

⑮ 発明の名称 乗算器

⑯ 特 願 平2-70678

⑰ 出 願 平2(1990)3月20日

⑱ 発 明 者 飯 野 秀 之 神奈川県川崎市中原区上小田中1015番地 富士通株式会社内
⑱ 発 明 者 飛 田 秀 憲 神奈川県川崎市中原区上小田中1015番地 富士通マイコンシステムズ株式会社内
⑲ 出 願 人 富士通株式会社 神奈川県川崎市中原区上小田中1015番地
⑲ 出 願 人 富士通マイコンシステムズ株式会社 神奈川県川崎市中原区上小田中1015番地
⑳ 代 理 人 弁理士 井 桁 貞 一 外2名

明 細 書

1. 発明の名称

乗算器

2. 特許請求の範囲

加算段を複数に分割し、

各分割加算段を配列形乗算器で構成するとともに、

各分割加算段の出力を加算する Wallace ツリー構成の加算回路を備えることを特徴とする乗算器。

3. 発明の詳細な説明

(概要)

並列形の乗算器に関し、

演算速度を向上しつつレイアウト容易性の改善を図ることを目的とし、

加算段を複数に分割し、各分割加算段を配列形

乗算器で構成するとともに、各分割加算段の出力を加算する Wallace ツリー構成の加算回路を備えることを特徴とする。

(産業上の利用分野)

本発明は、乗算器、特に並列形の乗算器に関する。

一般に、並列型乗算器は、直一並列形乗算器に比べて回路規模が大きくなる難点があるが、演算速度の点で優れており、乗算用 L S I (large scale integrated circuit) などに多用されている。

並列形乗算器は、被乗数 (multiplicand) と乗数 (multiplier) の部分積を求める回路 (例えば AND 回路) と、部分和を求める回路 (例えば全加算器) とを一つの単位とし、これをアレー状に並べて構成したものが基本となる。

(従来の技術)

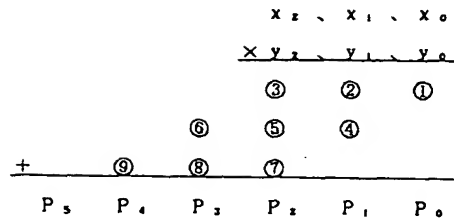
第5図はかかる並列形乗算器として公知の配列

形乗算器の概念図であり、10は被乗数 x_i (i は0、1、……)と乗数 y_i の部分積および部分和を求め、その総和(P_0, P_1, \dots)を出力する乗算網である。乗算網10には、第6図にその一部構成を示すように、全加算器(以下、FAと略す)からなる複数段(図では3段)の加算段が備えられる。

ここで、第6図の例では、便宜上3ビットの被乗数 x (x_0, x_1, x_2)および乗数 y (y_0, y_1, y_2)を扱う。最上段の加算段のFAに $x_i y_i$ の部分積(①~⑨)を入力すると、最下位段の加算段から部分和の総和($P_0 \sim P_3$)が取り出される。

すなわち、次の①~⑨の加算演算を実行する。

(以下、余白)



但し、

- ① $x_0 y_0$ 、② $x_1 y_0$ 、③ $x_2 y_0$ 、
- ④ $x_0 y_1$ 、⑤ $x_1 y_1$ 、⑥ $x_2 y_1$ 、
- ⑦ $x_0 y_2$ 、⑧ $x_1 y_2$ 、⑨ $x_2 y_2$ 、

このような配列形乗算器は、乗算網10を構成する単位回路(フルアダー)を規則正しく配置でき、レイアウト性に秀でた特長がある一方で、上位側の加算段の結果(SUM、CARRY)が順次下位段へと伝わる構成であることから、加算段の数(例えば乗数が27ビットならば27段)に相当して演算速度が遅くなる不具合を抱えている。

こうした不具合を解決するものとしてWallaceツリーが知られている。第7図はWallaceツリーの概念図である。Wallaceツリーは、被乗数 x_i と乗数 y_i の部分積を求める倍数器11および多数の全加算器をツリー状に接続した加算器12を有する。加算器12は部分積の各ビットごとの1の数を計数する。乗数 y_i を27ビットとすれば1の数の最大計数値は27となる。加算器12の中央部が最も多段となり両端になるにしたがってその段数を減少する。ビットの数に対する加算器12の最大段数は表1から求められ、例えばビットの数が9であれば、第8図に示すように、全加算器FAの段数が4段となる。

したがって、上述の配列形乗算器の場合では、その最大段数がビットの数と同じであったが、このWallaceツリーではビットの数より少ない段数となるから、段数差に相当して高速動作させることができる。

(発明が解決しようとする課題)

しかしながら、かかるWallaceツリーにあっては部分和のキャリCが上位側の部分和へと伝えられる構成となっていたため、キャリ伝達用配線とサム伝達用配線とが多数箇所で見交差し、設計が面倒でレイアウト性に劣るといった問題点があった。

本発明は、このような問題点に鑑みてなされたもので、演算速度を向上しつつレイアウト容易性の改善を図ることを目的としている。

(課題を解決するための手段)

本発明は、上記目的を達成するために、加算段を複数に分割し、各分割加算段を配列形乗算器で構成するとともに、各分割加算段の出力を加算するWallaceツリー構成の加算回路を備えることを特徴とする。

(作用)

本発明では、配列形乗算器で構成される分割加

算段ごとの演算結果が、Wallaceツリー構成の加算回路で加算される。したがって、加算段の分割数を適当に設定すれば、配列形乗算器およびWallaceツリーの双方の特長が生かされ、乗算器全体の演算速度の向上およびレイアウト容易性の改善が図られる。

〔実施例〕

以下、本発明を図面に基づいて説明する。

第1～4図は本発明に係る乗算器の一実施例を示す図である。

第1図において、 x_i は被乗数、 y_i は乗数を表す。これらの数はたとえばIEEE規格の倍精度浮動小数点表記の仮数部（ここでは、53（52ビット+かくれビット1）ビットとして扱う）である。なお、乗数 y_i は2次のBOOTH DECORDERによって53ビットから27ビットへとビット短縮されているものとする。

20は乗算器である。乗算器20は、3つの分割加算段（分割数は一例）23、24、25のそれぞれを配

列形乗算器で構成する加算段21と、Wallaceツリー構成の加算回路22とを備える。

加算段21の各分割加算段23、24、25の一部23a、24a、25a、たとえば入力9ビット分に相当する一部の構成は第2図に示される。すなわち、最上段（1段目）の5つの全加算器（FA）、2段目の5つの全加算器および3段目の4つの全加算器をアレイ状に接続し、1段目に部分積（①～⑤）を与え、3段目から各分割加算段ごとの演算結果（ $P_0 \sim P_5$ ）を取り出す。なお、この実施例では3段目の全加算器のキャリを同一段の全加算器に伝えないようにしている。こうすると、キャリ伝播遅延を改善できる。配列形乗算器の段数は、乗数 y_i のビット数に依存する。たとえば27ビットであれば27段となり、あるいは3ビットであれば第2図のように3段となる。すなわち、本実施例のように加算段21を3つの分割加算段23、24、25に分けることで、各分割加算段の段数を少なくでき、分割加算段ごとの遅延段数を減少できる。

各分割加算段の演算結果は加算回路22に入力さ

れる。ここで、加算回路22の一部の構成は第8図を参照する。第8図において、最上段（1段目）の3つの全加算器（FA）、2段目の2つの全加算器、3段目の1つの全加算器および4段目の1つの全加算器をツリー状に接続し、1段目に、各分割加算段23、24、25からの演算結果を与え、4段目からサムSおよびキャリCを取り出す。2～4段目には下位けたからのけた上がり（キャリ）が伝えられるとともに、上位けたへのけた上がりを取り出される。Wallaceツリーのキャリ配線とサム配線の交差点数は全加算器の段数に応じて増加する。すなわち、交差点数を少なくしてレイアウトの容易化を図るには、段数を減少すればよい。本実施例では、加算段21を3分割し、各分割加算段23、24、25ごとの演算結果を加算回路22に取り込むようにしている。これにより、ツリー段数を少なくでき、交差点数を減少してレイアウト容易性を向上できる。

このように、本実施例では、乗数 y_i のビット数に相当する演算段数の加算段21を複数分割し、

各分割加算段23……を配列形乗算器で構成するとともに、各分割加算段の出力を加算するWallaceツリー構成の加算回路22を備えたので、加算段21の分割数を適当に設定すれば、配列形とWallaceツリー形の双方の利点を活かすことができ、レイアウト容易性と動作速度とを共に向上することができる。

ちなみに、表2は加算段21を0分割（番号#0で示す）から27分割（番号#27で示す）までした場合の各遅延段数の一覧表である。#0は従来のWallaceツリーだけの場合に相当し、#27は従来の配列形乗算器だけの場合に相当する。そして、#0および#27以外が本実施例の乗算器20に対応する。表2において、乗算網で計算する段数は、乗数のビットの数（27）と同数または27を越える数となる。乗算網部の遅延段数Aは#27すなわち配列形乗算器だけの場合に最大となり、#0すなわちWallaceツリーだけの場合に最小となる。一方、ツリーの段数Bは入力ビット数によって決まり（表1参照）、#27すなわち配列

形乗算器だけの場合に最小、#0すなわちWallaceツリーだけの場合に最大(7段)となる。これにより、遅延段数の合計(A+B)は、配列形乗算器だけの場合(#27)で最大(27段)、Wallaceツリーだけの場合(#0)で最小(7段)となり、これらの#0#27間の適当な分割数を選択すれば、配列形乗算器とWallaceツリーの双方の特長を活かして、動作速度の向上とレイアウト容易性の改善とを共に達成できる。

第3、4図は本実施例の倍数器21および加算器22に使用する全加算器(FA)の一例を示す回路図である。何れの例も3ビット入力abcの加算結果SおよびキャリCを出力する。第3図は13個のCMOS40~52を用いて構成した例で、合計で26個のトランジスタを使用する。なお、破線の経路を接続すれば、CMOS1個(41)を除くことができる。第4図は3つのナンドゲート60~62、2つのEXゲート63、64を用いて構成した他の例で、合計で32個のトランジスタを使用する。

(発明の効果)

本発明によれば、上記のように構成したので、演算速度を向上しつつレイアウト容易性の改善を図ることができる。

4. 図面の簡単な説明

第1~4図は本発明に係る乗算器の一実施例を示す図であり、

第1図はその概念構成図、

第2図はその分割加算段の一部の構成図、

第3図はその全加算器の構成図、

第4図はその全加算器の他の構成図である。

第5、6図は従来の配列形乗算器を示す図であり、

第5図はその概念構成図、

第6図はその乗算網の一部の構成図である。

第7、8図は従来のWallaceツリーを示す図であり、

第7図はその概念構成図、

第8図はその加算部の一部の構成図である。

21...加算段、

22...加算回路、

23、24、25...分割加算段。

代理人 弁理士 井 桁 真 一

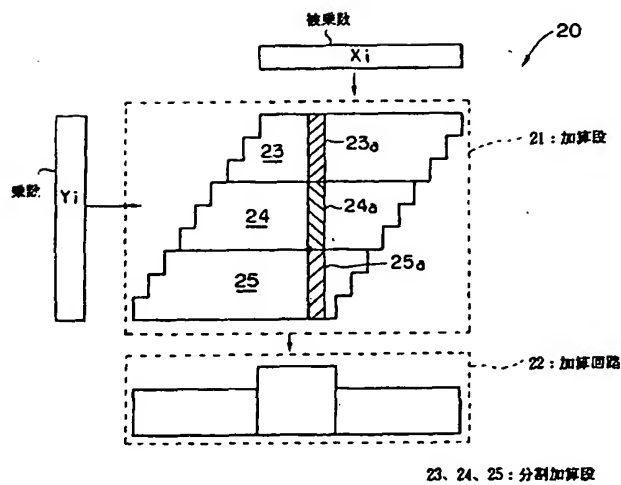
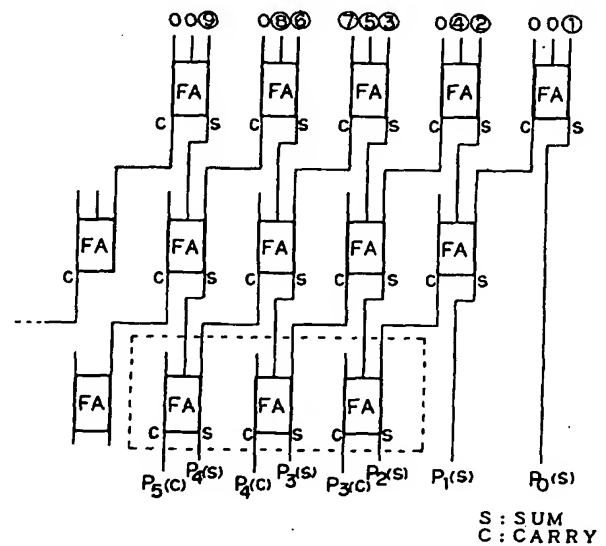


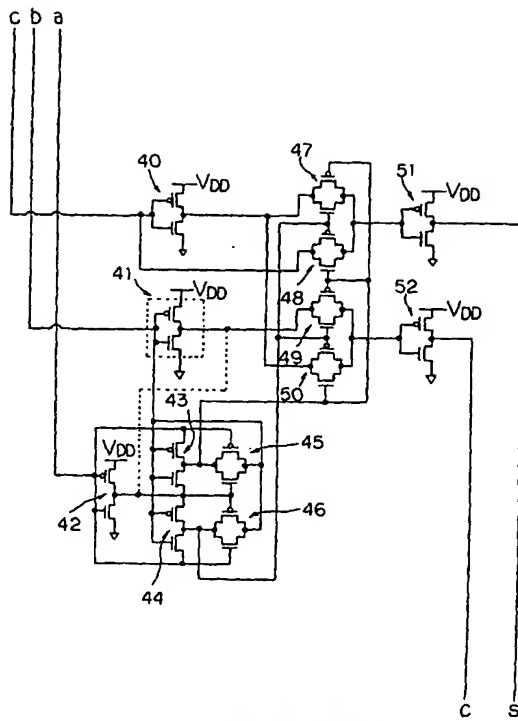
ビット数	3	4	5~6	7~9	10~13	14~19	20~28
段数	1	2	3	4	5	6	7

表 1

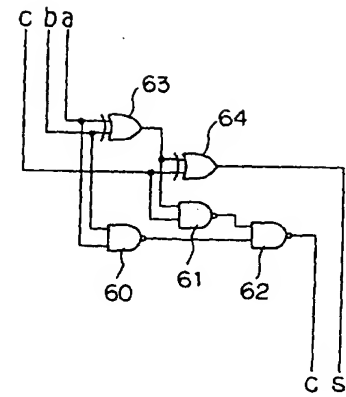
	A		B		A + B
	乗数網で計算する段数	乗数網部の遅延段数	Wallace ツリーの入力ビット数	ツリーの段数	遅延段数の合計
# 0	$1 \times 27 = 27$	0	27	7	7
\	\	\	\	\	\
# 2	$4 \times 7 = 28$	2	$7 \times 2 = 14$	6	8
# 3	$5 \times 6 = 30$	3	$6 \times 2 = 12$	5	8
# 4	$6 \times 5 = 30$	4	$5 \times 2 = 10$	5	9
# 5	$7 \times 4 = 28$	5	$4 \times 2 = 8$	4	9
# 6	$8 \times 4 = 32$	6	$4 \times 2 = 8$	4	10
# 7	$9 \times 3 = 27$	7	$3 \times 2 = 6$	3	10
# 8	$10 \times 3 = 30$	8	$3 \times 2 = 6$	3	11
\	\	\	\	\	\
# 27	$27 \times 1 = 27$	27	—	—	27

表 2

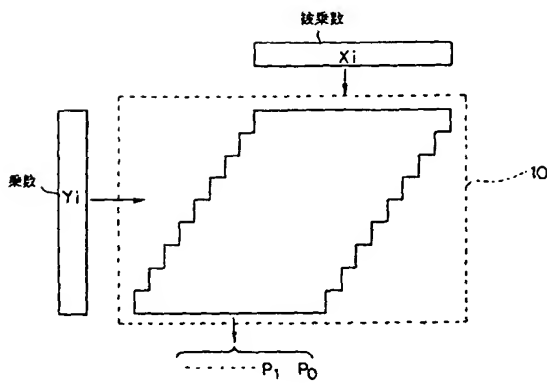
一実施例の概念構成図
第 1 図一実施例の分割加算段の一部の構成図
第 2 図



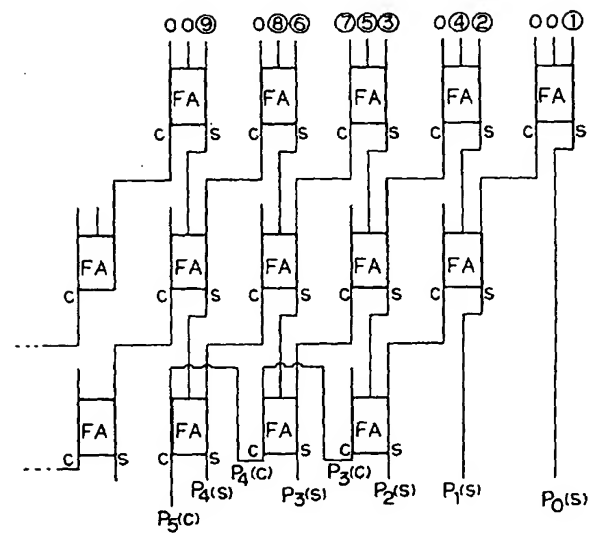
一実施例の全加算器の構成図
第 3 図



一実施例の全加算器の他の構成図
第 4 図

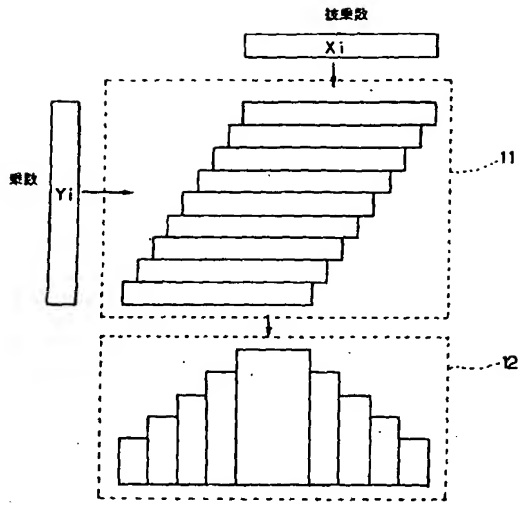


従来の配列形乗算器の概念構成図
第 5 図

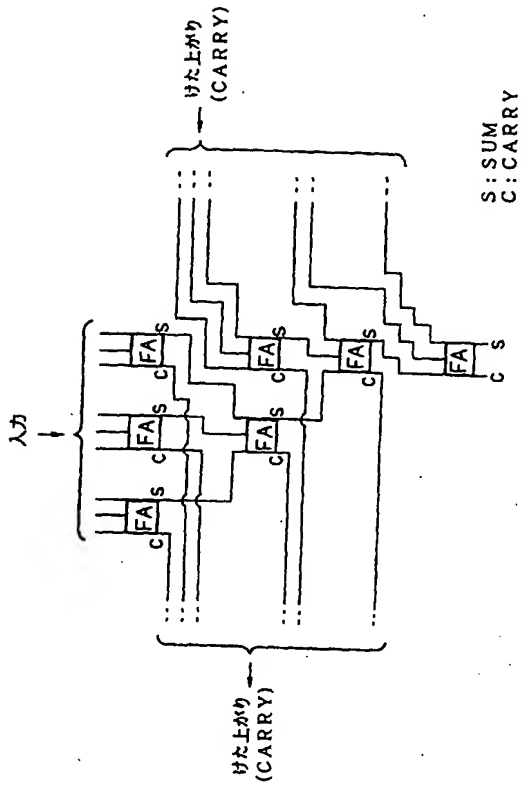


S : SUM
C : CARRY

乗算網の一部の構成図
第 6 図



従来のWallaceツリーの概念構成図
第 7 図



加算部の一部の構成図
第 8 図